Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 18 October 2018 doi:10.20944/preprints201810.0415.v1

Natural computing applied to the underground system: a synergistic approach for smart cities

Clemencio Morales Lucas · Luis Fernando de Mingo López · Nuria Gómez Blas

Abstract The management and proper use of the Urban Public Transport Systems (UPTS) constitute a field as critical as little investigated according to its relevance and urgent idiosyncrasy within smart cities realm. In this paper, a newfangled approach by using the Natural Computing paradigm and Collective Computation is shown, more concretely taking advantage of an Ant Colony Optimization algorithm variation in order to build a system that makes the complete control of the UPTS a tangible reality.

Keywords Urban public transport system · smart city · natural computing · collective computation · ant colony optimization.

1 Introduction

Since its pioneer conception in 1829, the underground trains have changed in such a mastodontic number of ways that it will take a long time to enumerate them. From the 47 km/h that Stephenson's train reached in the aforementioned year, to the 310 km/h that Spanish AVE is capable of obtain, trains have experimented an evident impact regarding their technology. However, these changes have not been applied to the management system and conception of the un-derground itself as it is nowadays. On the one hand, the rapidly-growing massification of the world's urban cores, in collaboration with the underground intensive use by citizens, is pushing the transition of these cores to the smart city purest concept, where every single element within the city has ratiocination enough for it to be called intelligent. In the year 2050, 66% of the world's population is expected to be living in urban cores (United Nations, Department of Economic and Social Affairs), increasing the current percentage of 54% in a 12%. In other words, the current estimations show that the continuous urbanization process that the world is facing, along with the overall growth of the world's population, will add another 2.5 billion people to urban populations by 2050, with close to 90% of the increase concentrated in Africa and Asia, according to a new United Nations report. To sum up, 66% of a world population of 9 Billion (5.94 Billion) will be living in urban cores in 2050. The aforementioned massification can be seen in figure 1.

On the other hand, it is important to note that this need has been outlined by organ-isms such as C.E.O.E (Spanish Confederation of Business Organizations). In fact, as described in CEOE (2015):

This frame of sustainability and efficiency that must involve the Smart Cities, has a direct relationship with other key areas, such as [...] the efficient management of the mobility of people [...] [Cities are lacking] Indicators for the collection appropriate measures [...] [Cities systems need] real-time knowledge about incidents, and an improved efficiency and management of the public transport. It is therefore evident that cities nowadays

Clemencio Morales Lucas
Escuela Técnica Superior de Ingeniería de Sistemas Informáticos, Universidad Politécnica de Madrid, Alan Turing s/n, 28031 Madrid, Spain
E-mail: clemencio.morales.lucas@alumnos.upm.es

Luis Fernando Mingo López
Deptartamento de Sistemas Informáticos, Escuela Técnica Superior de Ingeniería de Sistemas Informáticos, Universidad Politécnica de Madrid, Alan Turing s/n, 28031 Madrid, Spain
E-mail: fernando.demingo@upm.es

Nuria Gómez Blas
Deptartamento de Sistemas Informáticos, Escuela Técnica Superior de Ingeniería de Sistemas Informáticos, Universidad Politécnica de Madrid, Alan Turing s/n, 28031 Madrid, Spain
E-mail: nuria.gomez.blas@upm.es

© 2018 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 18 October 2018 doi:10.20944/preprints201810.0415.v1

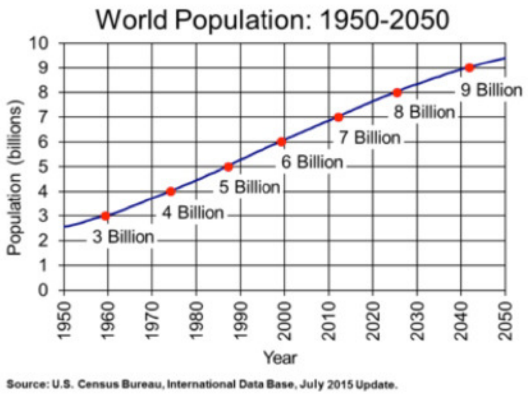

Fig. 1 World Population estimated growth between the years 1950 and 2050 (Source: United States Census Bureau, International Data Base).

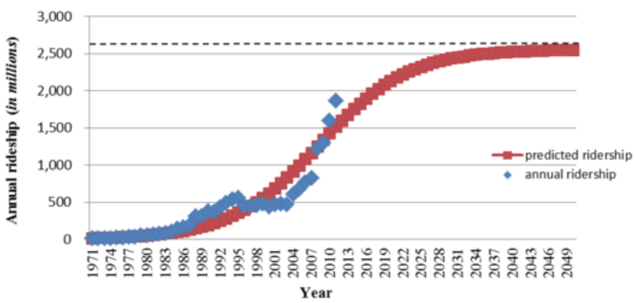

Fig. 2 Beijing Subway estimated growth between the years 1970 and 2050 (Source: Annual Subway Ridership of the Metropolitan Transportation Authority).

need a deep improvement on their IT systems and infrastructure, evolving to new schemes where data is seen as a binder for the city. In order to contribute to this goal, a gathering and management system mainly based on Natural Computing is presented on this paper.

Even the concept of Smart City is still being under constant redefinition, most authors agree that many different individuals, agents and devices, operate with their environment within the Smart City realm. Therefore, as Hollands (2008) points out, the relation among all of these elements will define the behaviour of the Smart City itself. It is easy to realize that an important area of the Smart City will be based in the interaction between the different components of it with their environment. This fact disembogues in a Socio-Collective Interaction, where the smart city in general terms, and specially the underground system beneath, can be seen as a huge swarm, where agents collaborate between them. The aforementioned approach justifies the present investigation project, mainly based on a change in the way of tackling the management processes of any underground system, using Collective Computation algorithms (Yi 2016, Chia 2014, Kaveh 2015) instead of the classical, graph-oriented ones (Trudeau, 1994).

2 Definition of the problem

The underground system beneath any urban core is a living, constantly in change entity. According the Annual Subway Ridership of the Metropolitan Transportation Authority (Metropolian 2016), 3.410 billion itineraries were made last year within Beijing under-ground system, more than the double of New York city subway itineraries, 1.763 billion. This number is massive at the time being, but still, it is expected to increase drastically in years to come. If we study the number of passengers using Beijing subway, the approximately 1800 million passengers/year it has nowadays, is expected to increase in 700 million by the year 2050. In other words, 27% of the world population will be using Beijing subway by 2050. The aforementioned increase in Beijing's subway usage is shown in figure 2

Thus, how can be these itineraries traced, letting the management know who is using the underground and when? How can the users rapidly know if there is an emergency or a path, which is not working due to technical errors within the underground? How can we analyze the massive data that can be potentially generated by so many itineraries? The response to these questions is precisely the reason that justifies the investigation, which aims to create a synergy of elements achieved due to the application of many newfangled Computation Paradigms (Fesharaki 2016, Majumdar 2013). These paradigms, in collaboration with strict software of control purposes, which will operate with user's Smartphone's, will head to an increase in the intelligence of the Urban Public Transport Systems (UPTS hereinafter).

Regarding the law-related aspects, it is important to remark that, according to the Regulation 2016/679 (EU, 2016) of the European Parliament and of the Council of 27 April 2016, dissociated data (those information related to a physical person that does not allow his direct identification) can be used in these sort of systems. As it is frequently seen and widely accepted in social networks, users will accept the share of their dissociated data by using the application for statistical, non-commercial purposes. This acceptance is shown in examples such Facebook and WhatsApp, that hold 1,871 and 1,000 million users, respectively (Statista, 2016). However, they will have the right to reject this safe data sharing. Rights of Access, Rectification, Cancellation and Opposition regarding dissociated data, as well as any kind of data stored, will be granted at any

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 18 October 2018 doi:10.20944/preprints201810.0415.v1

given time as in Spanish Organic Law 15/1999, December 13th (LOPD, 1999) which is aligned with European Union (EU) laws. Moreover, Data Suppression, Limitation and Portability will be always granted as well, as data storage good practices suggest in EU.

3 Investigation goals

The present investigation had the objective of fixing, chiefly, the following goals, that define an accurate overview of the investigation:

- Investigate the Computing Paradigms according to the realm of the Collective Collaboration: As it will be explained in further sections within this document, the Natural Computation stands as the best ally when it comes to this investigation aspect (Alaimo 2016). Genetic Algorithms (Holland 1992), Ant Colony Optimization (Dorigo 1992), Swarm Computing (Kennedy 1995), Grammatical Evolution (Ryan 1998) and Grammatical Swarm (O'Neill 2006), which have been widely investigated in order to find possible improvements, if any, in parallel investigations about algorithms and optimization.
- Find a nexus between the Computing Paradigms involved and the problem to solve: Once a strong theoretical overview has been given to the reader, the union point and nexus with the chased system will be described. Please bear in mind that, at the time being, there is no application of these algorithms in the UPTS context, factor that increases the newfangled character of the present investigation.
- Design and development of a system that, using the needed paradigms within Natural Computation, allows a wide study of the behaviour of the underground users: In a nutshell, the system aims to become a tool that makes possible the study of the user's behaviour, by taking dissociated data up in order to guard the privacy of the citizens. This objective will be possible thanks to the UPTS users Smart phones, for which an application is to be developed in the Android Operating System. Please note that this system will make possible to:
- Make precise studies about the statistical population that uses the UPTS. The dissociated data provided by commuters is a constantly updated and accurate source of information about the most frequent profiles in the UPTS. This information can be used as feedback for the system itself, establishing the exact number of people using it.
- Know, in an accurate way, the most popular routes for the users, as well as their behavior between the UPTS. It is important to note that this factor constitutes an open door for an efficient [1] management within the system. The study of the most popular routes can be seen as a first step in order to increase the frequencies of the highly crowded routes as needed.
- Prepare, thanks to the estimations gathered from the statistical study of the data, the UPTS in order to deal with peaks. Such scenario can be predicted by attending at atypical values within the data set gathered by the system.
- Detect anomalous situations, such a blocked train within a tunnel, or different scenarios where the number of users standing at the platform is high enough to fear an accident, surpassing the capacity of the specific dock.
- Prepare alternative routes in case of intensive use and/or fault of the UPTS systems. The system can detect whether a route is too crowded or not, allowing the UPTS to prepare alternative routes, if possible. The same solution applies in case of system fault.
- One of the main virtues of the system lies on it high level of customization, thus more functionalities and features, currently in an evaluation and delimitation stage.

4 Investigation goals

The system to be developed formed by essentially five groups of elements, has the main structure that can be seen in figure 3. There are five groups of elements within the system, which are described below:

- A: Fixed Smart dust: In Wireless Sensor Networks terms, this smart dust element, that will be unique in each UPTS station, will behave as the sink node. This special device will be integrated in the dock itself, receiving information from its Bn counterparts. The sink node will be the only element able to establish communication with the Operations and Control Center, D, and the elements within the C set.
- B[1,2,3,...,n]: Mobile Smart dust: The present element of the system, embedded in the UPTS trains fleet, will behave as a slave of the sink smart dust. They will establish communication with the sink, and will receive data from the elements within the C set, which will be explained below.
- C[1,2,3,...,n]: Users smart phones: This fundamental element within the system will be used by the

[1] It is important to note that the term efficient differs to effective in a subtle, but crucial manner; while an effective system achieves every objective, an efficient one achieves every objective as well but in the best, optimal way.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 18 October 2018 doi:10.20944/preprints201810.0415.v1

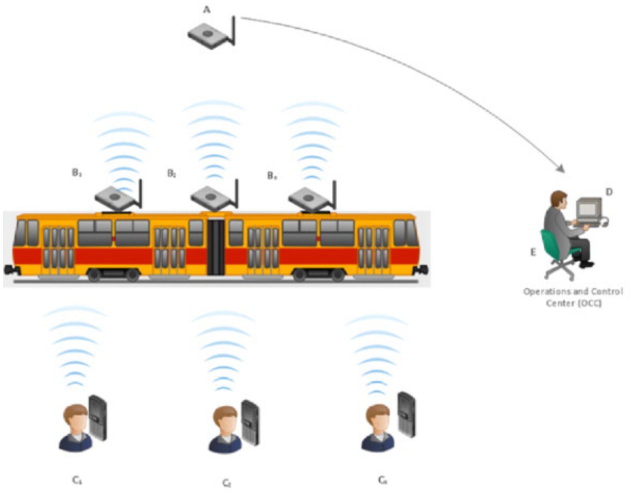

Fig. 3 Overview of the currently-under-development system.

users in order to make evident their presence in the dock. The elements within this group will be able to communicate with the sink smart dust, A, as well as with its counterparts in B, the mobile smart dust devices, that will make possible to know the number of users in the train. Note that the set formed specially by A and B[1,2,3,...,n] will shape the Wireless Sensor Network of the system, that will operate closely with the C[1,2,3,...,n] devices. A mobile application will be developed for the elements in C, that will retrieve the dissociated data of the users, let them know different routes in case of massive congestion, configure itineraries and show warning regarding abnormal situations that may occur within the UPTS.

- D: Operations and Control Center (OCC): This element will behave as the management point within the system realm, receiving the data sent by the smart dust in every single station, showing the pertinent status and the presence, if appropriate, of abnormal situations from the safety/systems failure point of view.
- E: System Administrator: Evaluator of the data showed by the OCC. Will operate accordingly to the UPTS current status and its environment, whether triggering a specific security protocol against failure or solving the different spurious scenarios that may occur.

5 Algorithms involved

Despite its apparent disparity, the following Computation Paradigms and the Algorithmic Techniques described below fall within the spectrum of the Natural Computation Paradigm. As long as the investigation is currently on a medium stage, the nexus with the system of some of these paradigms, as well as their application to the system, are still being under investigation.

As the accustomed reader will surely intuit, the algorithmic entities attached to this paradigm have, as its main base, the logic associated to phenomena present in nature, as well as the logic associated with the genetic-molecular base of the living beings, thus. As it can be read in Handbook of Natural Computing (Rozenberg, Back, Kok Joost, 2012), we can formally define Natural Computing as the set of computing techniques that circumscribe to, at least, one characteristic defined within the following group:

- Obtain its base from observing nature, establishing a computing simile.
- Base its reasoning in the use of the computers in order to synthesize natural.
- Use natural materials, from the logic or physical point of view, like DNA strings or chromosomes, in order to achieve its computational processes.

5.1 Genetic Algorithms

As stated by Charles Darwin in his opus magnum On the Origin of the Species (Darwin 1859), from immemorial times living beings have been forced to a continuous evolutive process looking for survival. Every single specie evolutes from a common antecessor looking for the adaptation to its environment and survive, following the process named natural selection. In a parallel way, Genetic Algorithms (GA hereinafter) follows the same pattern, trying to evolve a population. Thus, as it can be extracted from John H. Holland's Adaptation in Natural and Artificial Systems (Holland 1992), a GA can be formally defined as a set of ordered instructions, that aims to achieve an specific problem, which are based on the genetic-molecular base of the evolutive process of the living beings. It is remarkable that, despite the paternity of the GA is attributed to Prof. Holland, his sublime work means the colophon to the investigator cycle started by the distinguished Gregor Mendel in 1865, with his laws stated in Experiments in Plant Hybridization (Mendel 1965), based on the investigation over Pisum Sativum. In his publication, Mendel describes, using this specific pea variation, the basic rules related to the characteristics transmission between individuals through genetic inheritance. Actually, a GA has the objective of evolution certain specimens that set a population. In order to chase this goal, the GA uses random operations that establish a simile with the natural processes related to biological

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 18 October 2018 doi:10.20944/preprints201810.0415.v1

Fig. 4 Genetic operator crossover in a point between one-byte alleles (Source: University of Amsterdam – Faculty of Sciences, Department of Computer Sciences).

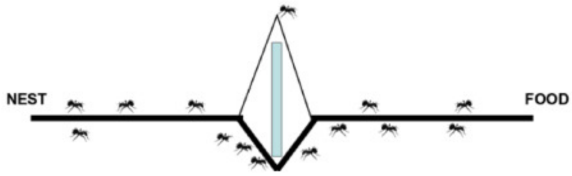

Fig. 5 Goss Experiment representation (Source: BioDat Research Group - University of Karlovo, Prague).

evolution. These methods, called genetic operators, are the following:

- Selection: In this operator, the GA chooses individual genomes from the population in order to start a later breeding process. Selection can be made by means of various techniques, as seen in A Comparison of Selection Schemes Used in Evolutionary Algorithms (Blickle, Tobias; Thiele, Lothar 1996). These techniques can be Roulette-Wheel Selection, Selection by Truncation, Selection by Ranking or Selection by Tournament, to quote a few of them.
- Crossover: Process whereby a variation in the chromosomes is done from a generation towards the following one. It is remarkable that, following the natural simile, the crossover mocks the sexual reproduction of the living beings. Letting a binary string be the information to be represented, there are several crossover techniques, and they all produce permutations in the chromosome. Seeing the chromosome as a set of alleles, the technique of crossover in a point can be an illustrative example; as shown in the following figure, once a bit within the chromosome is selected, every successive allele is exchanged between a chromosome and its pair, generating a new offspring in the process, see figure 4.
- Mutation: Variation within the genotype of a living being. Represents the action of the mutagens present in the ecosystem. It is remarkable that the genetic unit able to mutate is the gene, atomic, inheritable unity of data that builds up an individual's DNA.
- Recombination: Process whereby a DNA portion is cleaved in order to provide its further union to a different genetic material molecule. It is important to note that this action provokes different genetic permutations in a specie regarding its predecessors, producing chimeric alleles. This advantage makes the sexual reproduction possible between living beings, while avoiding Muller's ratchet [2].

5.2 Ant Colony Optimization

As Marco Dorigo and Gianni Di Caro establishes several times along (Dorigo, Di Caro 1999), Ant Colony Optimization (ACO hereinafter) is the name that refers to a multi-agent paradigm where every agent's behaviour is inspired on the ant idiosyncrasy when searching for livelihood. The algorithms that fall within this classification are based in Goss Experiment, using an Iridomyrmex humilis colony. In this experiment, the ant nest is connected to a livelihood source by means of two different paths, where one is longer than its counterpart, as the following figure shows:

After allowing the ants to freely move themselves along the scenario, it can be seen that, after an initial moments, they always choose the optimal, shorter path to the livelihood source. It is remarkable that, as well, this experiment demonstrates that a route selection probability is directly proportional to the length difference between both paths.

After studying the results thrown by Goss Experiment, a question arises; How do all the ants know what is the shortest path? The answer to this question is based on the concept known as stigmergy. The aforementioned concept alludes to those collaboration protocols, through the physical medium, where the different components collaborate due to the accumulation of objects or magnitudes in the environment, such pheromones or humidity. This concept is, precisely, the main tool within ant's communication; as the ants go back and forwards to the livelihood source, they deposit a chemical substance called pheromone. As it happens in several species, this substance provokes specific reactions and behaviour in the individual counterparts, allowing this, on this case, to know what the shortest path is.

It is remarkable that the directive that makes each ant k, placed in the i-th node, using a pheromone trail τ_{ij} in order to calculate the probability it has to use to chose a node j that belongs to N, as well as the following node where it has to move along, where N_i

[2] Named after its discoverer, Hermann Joseph Muller, is the process by which the different genomes of an asexual population accumulate deleterious mutations in an irreversible manner, that may result in the irrevocable extinction of the specie.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 18 October 2018 doi:10.20944/preprints201810.0415.v1

constitutes the set of nodes adjacent to i, is given by the equation:

$$p_{ij}^{k} = \begin{cases} \tau_{ij} & j \in N_i \\ 0 & j \notin N_i \end{cases} \quad (1)$$

5.3 Particle Swarm Optimization

Since the dawn of science, many scientific have been intrigued by a movement, as elegant as optimal, present in nature: The harmonious synchrony in bird flocks and fish shoals, where the individuals are able to move without even rub with each other, despite the hundreds, thousands of elements in certain cases, of individuals present in these sets. Thanks to scientific investigation, it has been demonstrated that, apart from this optimal movement, these animals present certain swarm patterns in their behaviour.

Concretely, it is important to highlight the hyperbolic interest of Grenander Heppner on their opus magnum A stochastic nonlinear model for coordinated bird flocks (Grenander, Di Caro 1999), where both zoologists synthesize their investigation referred to the nature-hidden directives that mark the asynchronous movement of the bird flocks, changing its direction suddenly in the presence of predators and tacitly regrouping, among other interesting abilities. In the same line, Reynolds Flocks, herds and schools: a distributed behavioural model (Reynolds 1987) stands out, aiming to the study of the interesting choreography that birds deploy.

Clustering the aforementioned references as base, the Particle Swarm Optimization (PSO, hereinafter) paradigm is known as the technique that pretends to optimize a problem due to a meta-heuristic strategy, which is, due to the iterative trial of improving a candidate solution with regards to a pre-stipulated quality criterion. Thus, in a way that reminds to GA, PSO optimizes a problem starting from a set of candidate solutions, typically particles over the space, moving them along through the searching space without forgetting the premises of PSO mathematical base, which involves the position and the speed of the particles. As it can be inferred, the technique mimetizes the group behaviour of the aforementioned living beings, where each individual movement is influenced by the best local position known, while, in a parallel way, the swarm maintains a best global position known. This best global position is updated by the best position known by all the individuals in the swarm, fact that will guide the set to move searching for the best global position.

PSO adopts a tiny number of postulations along its execution process, exploring a mastodontic search space. Despite from that, PSO is a meta-heuristic, so it is not possible to adamantly ensure that the algorithm is going to find an optimal solution of the problem for every single case. In a more mathematical, accurate way, PSO does not use the gradient of the tackled problem, which means that this technique does not require the problem to be differentiable, as well as it happens in typical optimization methodologies such Cuasi-Newtonian methods or Gradient Descent. Thus, PSO can be used, enjoying a high success rate, in optimization problems that are especially non-regular, where there is certain ambient noise, or those presenting a dynamic, changing-over-time behaviour. PSO algorithm pseudocode can be stated as following:

Let the following symbols represent properties of a particle:

- x_i is the current position of particle i
- v_i is the current velocity of particle i
- p_{Best} is the personal best position of the particle
- g_{Best} is the global best particle
- $c_1 \in R$ is the personal influence (acceleration coefficient)
- $c_2 \in R$ is the global influence (acceleration coefficient)
- r_1 and r_2 are random numbers distributed using a uniform pattern on interval $[0, 1]$.

With these notations, the formula to calculate a particle's velocity at time $t + 1$ (time is simulated using iteration number) is:

$$\begin{aligned} v_i(t+1) = v_i(t) + & \\ & + c_1 * r_1 * (p_{Best} - x_i) + \\ & + c_2 * r_2 * (g_{Best} - x_i) \end{aligned} \quad (2)$$

where, r_1 and r_2 are randomly generated for every velocity update and $0 \leq r_1, r_2 \leq 1$. They should both be different each iteration. And c_1, c_2 are user defined values called acceleration coefficients where $0 \leq c_1, c_2 \leq 2$. Their value depends on the problem to be optimized.

New particle position at time $t + 1$ just adds the newly calculated velocity to its current position at time t. In other words, the position now is the previous one adding its velocity, see figure 6. Note that g_{Best} refers to a star topology.

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (3)$$

Algorithm 1 shows the standard PSO process to update location and velocity of particles.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 18 October 2018 doi:10.20944/preprints201810.0415.v1

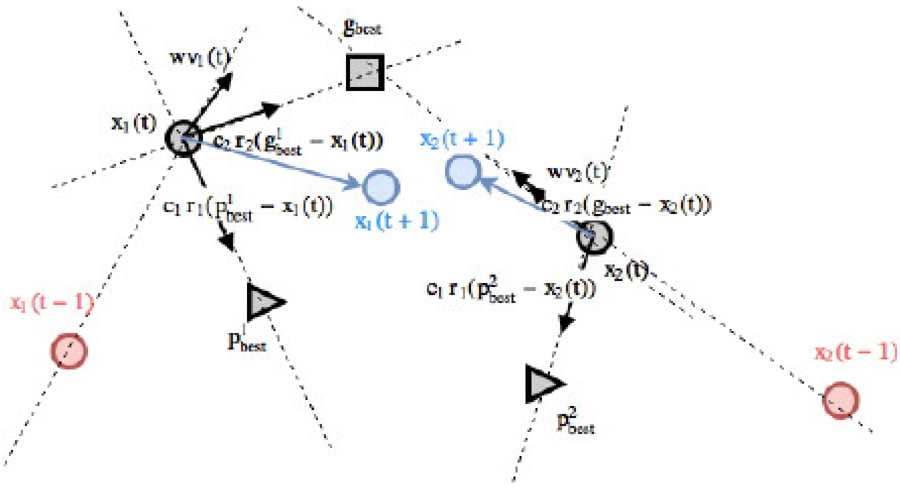

Fig. 6 Particle swarm optimization model: position update using a star topology (g_{Best}) and an inertia w term.

Algorithm 1 Standard particle swarm optimization algorithm: update process.

```
for Each time step t do
    for Each particle i in the swarm do
        update position x_i(t+1) using equations (2, 3)
        calculate particle fitness f(x_i(t+1))
        update p_Best and g_Best
    end for
end for
```

5.4 Fireworks Algorithm

As history shows, mathematical optimization has always been a field under investigation (Boyd, 2004). More concretely, and due to its vital importance on Computer Science, the search processes involved in computing problems have been widely investigated. These processes are highly related to sorting algorithms too.

A search algorithm is a set of mathematical instructions aiming to place a custom order between the elements within a collection. Optimization techniques, as well as efficient sorting, has been the main objective when it comes to these kind of algorithms since the first investigations on the field (Cormen, 2001).

In a similar way, Fireworks Algorithm (FWA onwards) belongs to the Swarm Intelligence spectrum and takes its inspiration by observing fireworks explosion (Tan, 2015). This algorithm is proposed mainly for the optimization of complex functions, and its implemented following an accurate simulation of fireworks explosion process. FWA takes special attention to keep the diversity of sparks as it will be explained, by maintaining two different search processes. In general terms, FWA constitutes an approach to explore a massive search space. This search is based on the search of random points confined by a certain distance measure that hopes one or more of the points of interest will yield promising results. Once these points give interesting results in terms of the mathematical function to be optimized, a more concentrated search will be spawned in the near points, iterating through the algorithm until an optimal solution is found.

Regarding FWA operating mode, the algorithm starts with the selection of a concrete number, N, of initial locations. N fireworks will be thrown at these locations, and the fireworks will consequently throw sparks. The location of these sparks will be retrieved to evaluate its quality, and in case the optimal solution has been found, the algorithm will finish. Otherwise, FWA will set another N fireworks at N locations, spawning and iterating over and over again over the aforementioned process until the optimal solution is found.

The following image shows the flux diagram of FWA (Tan, Y et al, 2015):

In order to validate the convergence curves of FWA, Clonal Particle Swarm Optimization (CPSO) and Standard Particle Swarm Optimization (SPSO) are taken in (Tan et al 2015). A set of eight different benchmark functions (Sphere, Rosenbrock, Ellipse, Cigar, Rastrigin, Griewank, Tablet and Schwefel) averaged over 20 independent runs are thrown on these algorithms, reaching the conclusion that FWA presents a higher speed compared to CPSO and SPSO as the following figure shows:

The information shown in the last figure, with the addition of Ackley benchmark function, can be analysed in the following figure, where the Statistical Mean and Standard Deviation for FWA are far more agile compared to CPSO and SPSO:

To sum up, FWA finds brilliant solutions with 1000 times of function evaluations, reflecting the quick con-

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 18 October 2018 doi:10.20944/preprints201810.0415.v1

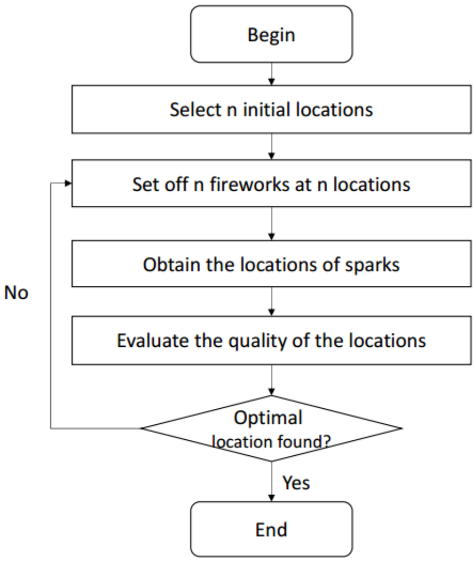

Fig. 7 FWA Algorithm Flux Diagram (Source: Introduction to Fireworks Algorithm. Ying Tan et al, Peking University, 2015.)

vergence speed of the algorithm against CPSO and SPSO. This algorithm is therefore a strong ally when it comes to smart cities environment applications in a computing-related way, being currently investigated in order to create a system capable of searching lost people in the smart city rural surroundings.

FWA has multiple applications, as (Ying, 2015) shows with a system for data mining low-rank applications such web search. These systems reduce storage and requirements at runtime, as well as the noise regarding data representation when it comes to essential associations. The Non-negative Matrix Factorization, whose parameters are widely explained at (NMF, (Lee et al, 1999)) arrives to a low-rank approximation that verifies that non-negativity constraints are satisfied. NMF approximates a data matrix by where and are the NMF factors. NMF requires all entries in, and to be zero or positive. The following figure shows the FWA-based optimization algorithm pseudo code for NMF –see figure 10–, showing explicitly the high level of configurability associated to FWA, where SIO stands for Swarm Intelligence Optimization.

6 Application to the smart cities realm and potential results

It is inferred from this paper that the wide spectrum of applications that can be extracted from Natural Computing and applied to the smart city is massive. Moreover, as it has been pointed out along the paper, the cities need deep changes in order to be called 'smart'. As long as in short terms Natural Computing mimics phenomena present in nature, it turns out to be an excellent ally when improving the different systems that belong to a city: A city can be seen as a swarm of individuals that operate among a system. Thus, following a bottom-up scheme, when it comes to a city's realm, the citizens can be seen as particles, therefore they conform a swarm, atomic work element of the Natural Computing paradigm. This simile opens the smart city concept to be widely improved by applying Natural Computing, where Genetic Algorithms, Ant Colony Optimization, Particle Swarm Optimization and Fireworks Algorithm specially stand out. It is important to point, though, that this paper has motivated parallel investigation lines in the authors, and thus another way of thinking when it comes to the particle-people simile is being under investigation. In this parallel approach, people that share the same information is being grouped in sets of particles, spawning the concept of a more complex type of 'super-particles' among the swarm. This concept is studied as well in (Chen H 2010).

On the one hand, the efforts under the current investigation are being currently driven into the ACO Algorithm spectrum: Even efficient, ant pheromone is simple, primitive; it only marks the shortest path to the livelihood source, but; what if this pheromone concept is extended to a super-pheromone? A super-pheromone will store dissociated data of a person (i.e.: age, gender, education level, etc), thus, it will be possible to know which person profile is transiting for each UPTS section by seeing the user as an ant. More concretely, by applying the schema shown in [Fig 3. Overview of the currently under development system] under the section [4. Topology of the system], the UPTS will improve its perception, knowing who is circulating where, and consequently showing publicity screens according the relevant information for the public. (For instance, it will be more effective to show the publicity related to a new videogame near an institute area when the train is crowded by young people, while a new credit card with

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 18 October 2018 doi:10.20944/preprints201810.0415.v1

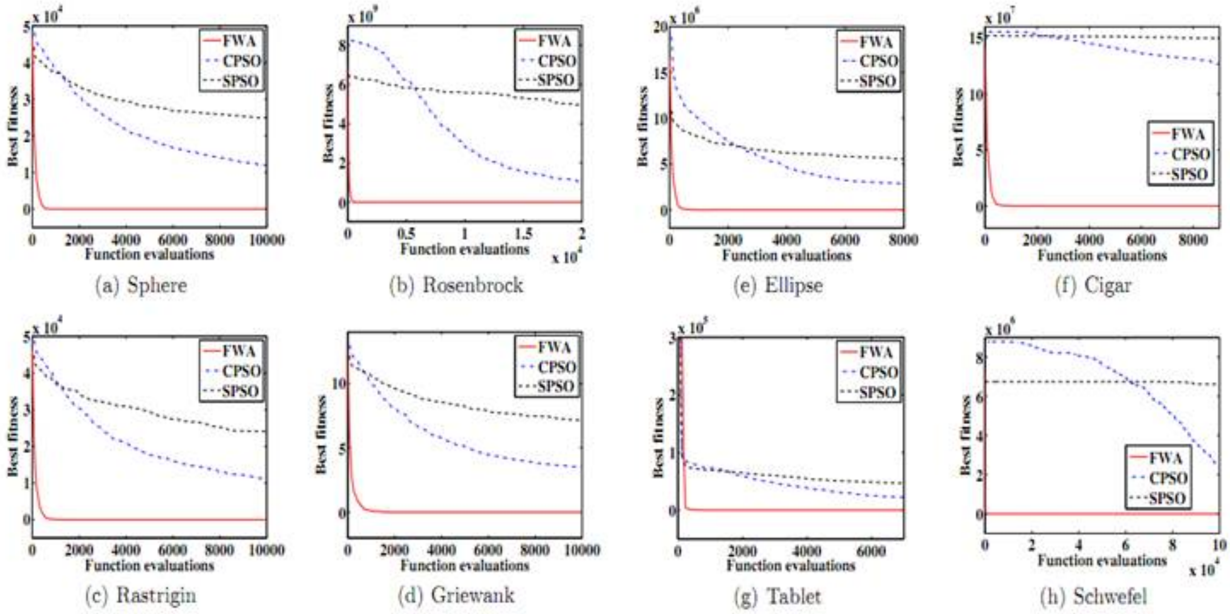

Fig. 8 Benchmark Functions results when comparing FWA, CPSO and SPSO (Source: Introduction to Fireworks Algorithm. Ying Tan et al, Peking University, 2015.)

Function	FA's mean (StD)	CPSO's mean (StD)	SPSO's mean (StD)
Sphere	0.000000 (0.000000)	11857.425781 (3305.973067)	24919.099609 (3383.241523)
Rosenbrock	19.38330 (11.94373)	2750997504.000000 (1741747548.420642)	5571942400.000000 (960421617.568024)
Rastrigin	0.000000 (0.000000)	10940.148438 (3663.484331)	24013.001953 (4246.961530)
Griewank	0.000000 (0.000000)	3.457273 (0.911027)	7.125976 (0.965788)
Ellipse	0.000000 (0.000000)	2493945.500000 (1199024.648305)	5305106.500000 (1117954.409340)
Cigar	0.000000 (0.000000)	122527168.000000 (28596381.089661)	149600864.000000 (13093322.778560)
Tablet	0.000000 (0.000000)	15595.107422 (8086.792234)	42547.488281 (8232.221882)
Schwefel	4.353733 (1.479332)	8775860.000000 (1217609.288290)	6743699.000000 (597770.084232)
Ackley	0.000000 (0.000000)	15.907665 (1.196082)	18.423347 (0.503372)

Fig. 9 Statistical mean and standard deviation of solutions found by the FA, the CPSO and the SPSO on nine benchmark functions over 20 independent runs of 10000 function evaluations.(Source: Introduction to Fireworks Algorithm. Ying Tan et al, Peking University, 2015.)

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 18 October 2018 doi:10.20944/preprints201810.0415.v1

1: Given matrix $A \in \mathbb{R}^{m \times n}$ and $k \ll min\{m, n\}$;
2: $H0 = \text{rand}(k, n)$;
3: `% Compute in parallel`
4: **for** $i = 1$ to m **do**
5:     Use SIO to find $\mathbf{w}_i^r$ that minimizes $||\mathbf{a}_i^r - \mathbf{w}_i^r H0||_F$, (min $||.||_F$ of row i of D);
6: **end for**;
7: `% Gather`
8: $W = [\mathbf{w}_1^r; \ldots; \mathbf{w}_m^r]$;
9: `% Compute in parallel`
10: **for** $j = 1$ to n **do**
11:     Use SIO to find $\mathbf{h}_j^c$ that minimizes $||\mathbf{a}_j^c - W\mathbf{h}_j^c||_F$, (min $||.||_F$ of col j of D);
12: **end for**
13: `% Gather`
14: $H = [\mathbf{h}_1^c, \ldots, \mathbf{h}_n^c]$;

Fig. 10 Pseudo code for the initialization procedure for NMF factors W and H. The two for-loops in lines 4 and 10 can be executed concurrently. (Source: Introduction to Fireworks Algorithm. Ying Tan et al, Peking University, 2015.)

certain bonuses will be more appropriate in the UPTS section beneath the financial area of the city.)

On the other hand, GA paradigm is being used as a way to evolve a route instead of a chromosome population: by means of a Smartphone application, users will be able to quickly know the best route between two points in the UPTS, as well as backup routes in case of systems breakdown. PSO can be used for studying the data retrieved in the Operations and Control Center (OCC). This will make possible to optimize the system by applying a statistical investigation over the data, detecting statistical outliers and acting in consequence. It is remarkable that the investigation regarding this slope and other Natural Computing paradigms is in an early stage, thus new applications are susceptible to emerge.

7 Conclusions and discussion

In this paper, a newfangled scheme for endowing intelligence to a city UPTS is given, chasing the transition of the city to a smart city. In this approach, Natural Computing paradigm will be applied to the system, after a deep investigation that aims to improve the involved paradigms, if possible. Despite the investigation is still being in an early stage, the system is likely to improve the data gathering related to the UPTS in a mastodontic way, allowing the pertinent authorities to improve the system and even monetize the information gathered by the system under development. Moreover, users will be able to enjoy a better use of UPTS, knowing alternative routes in case of systems breakdown and being able to travel in an efficient way.

References

References

1. Chia, T. Z., Chengb, H., Pagec, J.R., and Ahmed, N. A. (2014), Evolving swarm of UAVs, Advances in Aircraft and Spacecraft Science, An Int'l Journal 1 (2), 219-232.
2. Confederación Española de Organizaciones Empresariales (2015), Acciones prioritarias para el desarrollo de las Smart Cities en España, (Prioritary actions for the development of the Smart Cities in Spain) Comité de Smart Cities, Madrid.
3. Blickle, T. and Thiele, L. (1996), A Comparison of Selection Schemes Used in Evolutionary Algorithms, Evolutionary Computation, 4 (4), 361-394.
4. Darwin, C. (1859), On the Origin of Species, London.
5. Dorigo, M. and Di Caro, G. (1999), The Ant Colony Optimization Meta-Heuristic, New ideas in optimization. McGraw-Hill Ltd, 11-32.
6. Fesharaki, J. J. and Golabi, S. (2016), A novel method to specify pattern recognition of actuators for stress reduction based on particle swarm optimization method, Smart Structures and Systems, 17 (5), 725-742.
7. Grenander, L. and Heppner, C. (1990), A stochastic nonlinear model for coordinated bird flocks, The ubiquity of chaos, E. Krasner, 233-238.
8. Holland, J. (1992), Adaptation in natural and artificial systems, Cambridge, Mass., MIT Press.
9. Hollands, R. (2008). Will the real smart city please stand up?, Metropolitan Transportation Authority, Annual Subway Ridership, 12 (3), 303-320.
10. Kaveh, A. and Maniat, M. (2015), Damage detection based on MCSS and PSO using modal data, Smart Structures and Systems, An Int'l Journal 15 (5), 1253-1270.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 18 October 2018 doi:10.20944/preprints201810.0415.v1

11. Majumdar A., Maity D. and Maiti D. K. (2013), Damage assessment of beams from changes in natural frequencies using ant colony optimization, Structural Engineering and Mechanics, 45 (3), 391-410.
12. Mendel, G. (1965), Experiments in plant hybridisation, Cambridge, Harvard University Press.
13. Milazzo, A. and Orlando, C. (2016), Nonlinear model based particle swarm optimization of PID shimmy damping control, Advances in Aircraft and Spacecraft Science, 3 (2), 211-224.
14. Reynolds, H. (1987) Flocks, herds and schools: a distributed behavioral model. Computer Graphics, 21 (4). 25-34.
15. Rozenberg, G., Back, T. and Kok, J. (2012), Handbook of natural computing, Berlin: Springer.
16. Yi, T. H., Wen, K. F. and Li, H. N. (2016), A new swarm intelligent optimization algorithm: Pigeon Colony Algorithm (PCA), Smart Structures and Systems, 18 (3), 425-448.
17. Chen H, Zhu Y, Hu K (2010). Discrete and continuous optimization based on multi-swarm coevolution. Nat Comput, 9(5). 659-682
18. United Nations, Department of Economic and Social Affairs: "World's population increasingly urban with more than half living in urban areas" http://www.un.org/en/development/desa/news/population/world-urbanization-prospects-2014.html
19. Statista, The Statistics Portal: "Most famous social network sites worldwide as of January 2017, ranked by number of active users (in millions)" https://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/
20. LOPD: Spanish Organic Law 15/1999, December 13th, Personal Data Protection (Ley Orgánica 15/1999, de 13 de diciembre, de Protección de Datos de carácter personal)
21. Holland, J. (1992). Adaptation in Natural and Artificial Systems. Cambridge, MIT Press.
22. Dorigo, M. (1992). Optimization, Learning and Natural Algorithms. Politecnico di Milano, Italy.
23. Kennedy, J., Eberhart, R. (1995). Particle Swarm Optimization. Proceedings of IEEE International Conference on Neural Networks, Vol IV. 1942–1948. DOI:10.1109/ICNN.1995.488968
24. Ryan, C., Collins, J.J., O'Neill, M. (1995). Grammatical Evolution: Evolving Programs for an Arbitrary Language. Proceedings of the First European Workshop on Genetic Programming, Wolfgang Banzhaf, LNCS, 83-96.
25. O'Neill, M. and Brabazon, A. (2006) Natural Computing 5: 443. DOI:10.1007/s11047-006-9007-7, LCNS.
26. Tan, Y., Yu, C., Zheng, S., Ding, K. (2015) Introduction to Fireworks Algorithm. Computational Intelligence Laboratory of Peking University. Peking, China.
27. Cormen, H., Leiserson, E., Rivest, L., Stein, C. (2001) Introduction to Algorithms. Second Edition. MIT Press and McGraw-Hill.2-2, p.40.
28. Tan, Y. (2015) Fireworks Algorithm. A Novel Swarm Intelligence Optimization Method. Springer. DOI: 10.1007/978-3-662-46353-6. p. 17-35
29. Lee, DD., Seung, HS. (1999). Learning the parts of objects by non-negative matrix factorization. US National Library of Medicine National Institutes of Health. p. 43-67.